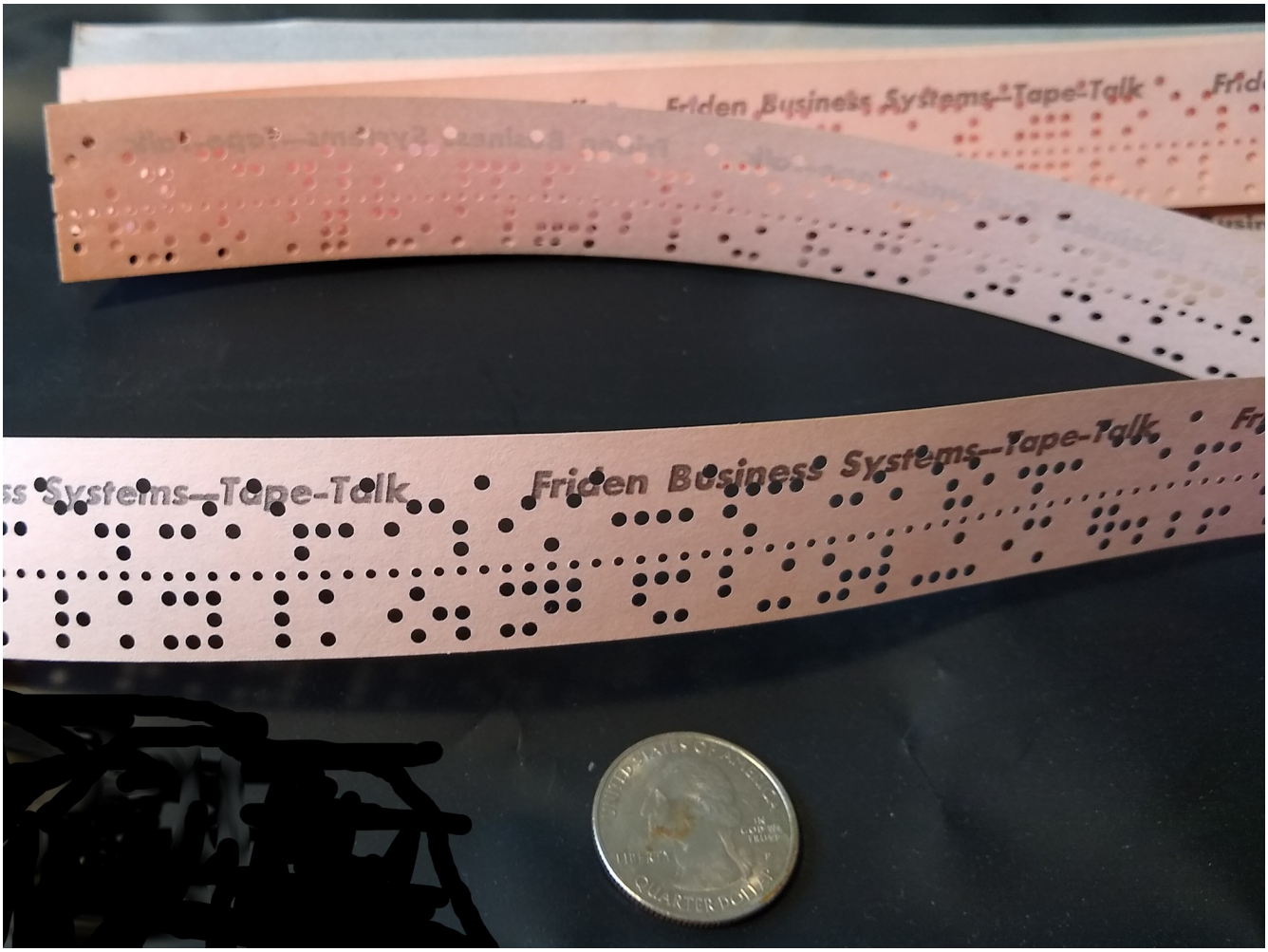


Digiac paper tapes were generated by the Punch Tape instruction and they could be read by the Read Tape instruction. This is 8-level tape with three bits below and five bits above the row of smaller feed holes. The least significant bit ia at the the lower edge. A bit is zero if the paper is intact; it is one if there is a hole present.

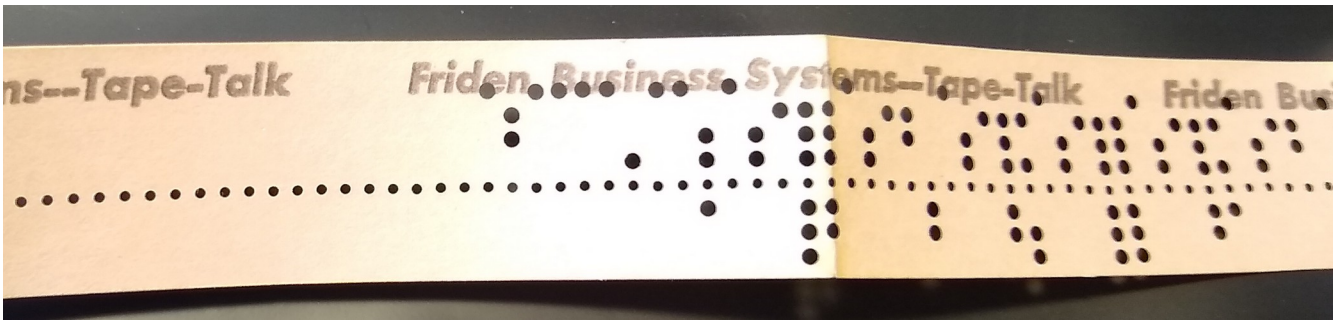
As oriented in the pictures, the tape would be read from the left to the right. My transcription of a tape generated one byte for each column. These bytes, in the sequence they were read, are the content of a .ptp file.

The Digiac paper tapes are binary. Columns with no holes were consider leader. When reading a tape, leader would be ignored and bypassed. My recollection is that a RT instruction would terminate if the reader ran past the end of the tape.

Digiac I/O instructions transfer whole words. Five tape columns were used to represent each 25-bit word. First the sign, then two octal digits at a time from high to low order are punched into the tape. Any column where the bits are all zero gets the 6^{th} bit set to have a value of 100(octal). A positive sign was encoded as 100, just like zero. I do not know the encoding for a negative sign.

The picture below shows the beginning of the data in stok.ptp. Reading from left to right in octal the column values and corresponding words are:

```
100-60-100-100-100   60000000
100-10-100-100- 34   10000034
100-30- 40- 77- 76   30407776
```



This particular tape contains two programs. The first program can be used to zero the memory locations used as a random seed for the stock market game; thus producing a repeatable game. The second program is the game itself. Contemporary instructions written at the start of the tape are:

```
STOK - No Randomize
0000 – 60000000 – AUTO
1400 -  Start/Restart
```

About 3 feet into the tape is another section of leader with the equivalent instructions for the game:

```
STOCK MARKET – STOK II
To Read In:    0000 - 60000000 - AUTO
               1400 - Start/Restart
```

You can use the `tape_dump.py` program to print out the contents of this tape. For example:

```
$ ./tape_dump.py tape/stok.ptp

File: tape/stok.ptp raw data length: 9890
removed 250 leading zero bytes
  Addr    Octal    Char
  ====  ========   ====
. 0000  60000000  |)000|
. 0001  10000034  |800L|
. 0002  30407776  |H.+"|
. 0003  30406006  |H.)6|
. 0004  30407653  |H."X|
…
```